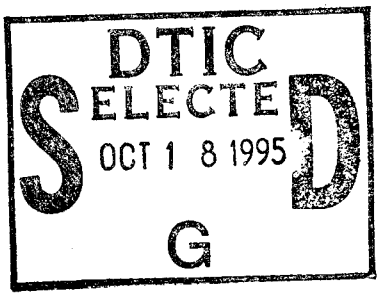
RL-TR-95-182
Final Technical Report
September 1995

ADAPTIVE SENSOR ARRAYS WITH SUBBAND SIGNAL PROCESSING

Georgia Institute of Technology

Dr. Mary Ann Ingram



APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Laboratory
Air Force Materiel Command
Griffiss Air Force Base, New York

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-95-182 has been reviewed and is approved for publication.

APPROVED: Richard N. Smith

RICHARD N. SMITH
Project Engineer

FOR THE COMMANDER: John A. Graniero

JOHN A. GRANIERO
Chief Scientist
Command, Control & Communications Directorate

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL (C3BA) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1995	Final Dec 94 - Jun 95

4. TITLE AND SUBTITLE

ADAPTIVE SENSOR ARRAYS WITH SUBBAND SIGNAL PROCESSING

5. FUNDING NUMBERS

C - F30602-94-C-0033
PE - 62702F
PR - 4519
TA - 63
WU - P5

6. AUTHOR(S)

Dr. Mary Ann Ingram

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Georgia Institute of Technology
School of Electrical & Computer Engineering
Atlanta GA 30332-0250

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Rome Laboratory (C3BA)
525 Brooks Rd
Griffiss AFB NY 13441-4505

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RL-TR-95-182

11. SUPPLEMENTARY NOTES

Rome Laboratory Project Engineer: Richard N. Smith/C3BA/(315) 330-7436

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

A tree-structured multirate filterbank is used for low-complexity space-time adaptive processing for space-based communications. The filterbank is used to improve convergence speeds in the Least Mean Squares (LMS) algorithm and to reduce processor complexity in a wideband adaptive array processor that performs interference cancellation. The filterbank concept reported here may be described as a hardware-friendly method of implementing a partially adaptive array, because the coefficients used in the linear combinations of (steered) array element outputs are only 1, -1, and 0. An algorithm for adaptively growing the tree-structured filterbank allows the partially-adaptive array to be time-varying, i.e., to change the subspace for the adaptive weights as the jamming scenario changes. It was observed, however, that a great majority of reasonable jamming scenarios do not require distinct tree structures. The singular value decomposition on a large sample of full-rank optimal weight vectors shows that, indeed, there is a fixed, partially-adaptive array that serves a large sample of jamming scenarios with negligible penalty relative to the optimum full-rank solution for each scenario.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Adaptive arrays, Interference cancellation, Subband decompositions	52
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Contents

List of Tables

List of Figures

1 Introduction

The subject of partially adaptive array processing has received significant attention in the late 70's and 80's [1, 9, 8, 10, 11]. Several of these papers assume a generalize sidelobe canceller (GSC) implementation of the linearly-constrained minimum variance (LCMV) beamformer, which allows unconstrained adaptive minimization of array output power when the array is constrained to provide an all-pass frequency response in given directions of arrival. The GSC decomposes the observation vector (snapshot of element outputs), which can contain thousands of entries, into a "constraint space" component which lies in the space spanned by the columns of the constraint matrix C, and a "noise space" component that is orthogonal to the constraint space component. As shown in Figure 1, the standard pro-

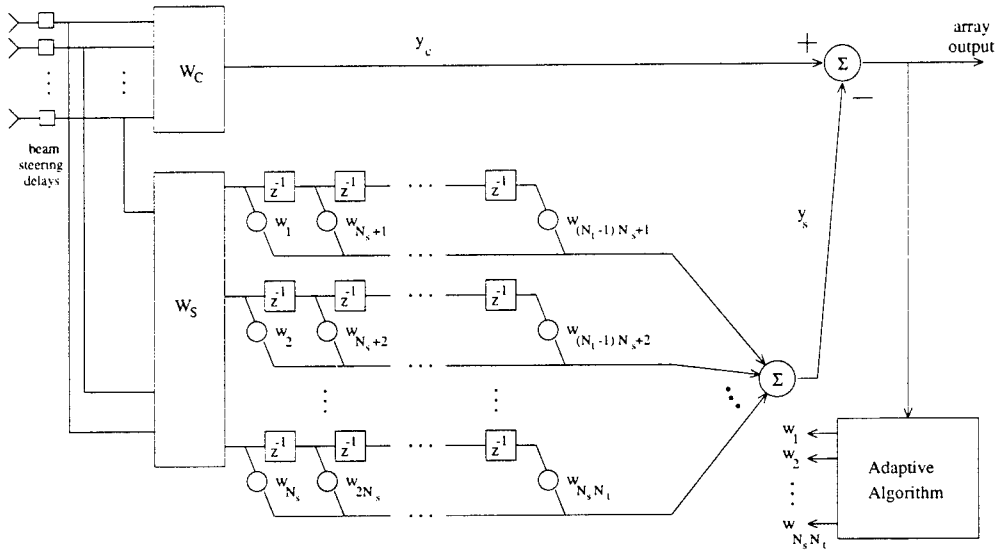


Figure 1: Standard generalized sidelobe canceller.

cedure is to apply adaptive weights, controlled by some adaptive algorithm, to the noise space component and subtract the result from the constraint space component to get the output of the GSC. The objective of a partially adaptive array design is to reduce the number of adaptive weights, since they are expensive to implement. Stated differently, the objective is to reduce the rank of the autocorrelation matrix for the observation vector (that is, the

vector to which the adaptive weights are applied). Partial adaptivity or rank reduction is achieved by operating on the noise space (column) vector with a non-square (fewer rows than columns) orthonormal transformation T, as shown in Figure 2, where the rows of this matrix form a lower dimensional

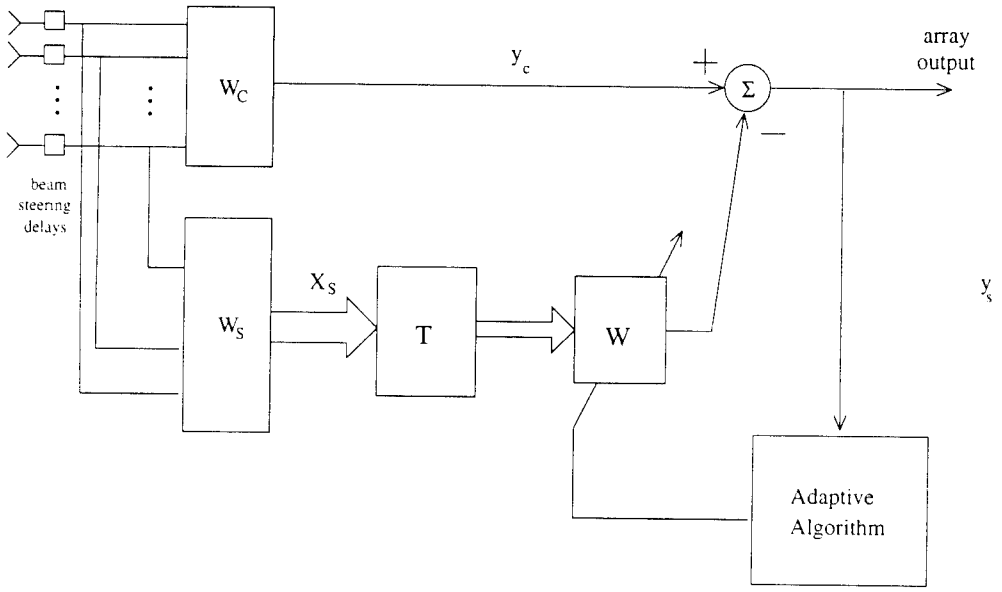


Figure 2: Transform domain

basis for the adaptive weight vectors. This lower-dimensional subspace may be called the *adaptive weight vector solution space* or AWVSS, after [8]. The problem becomes how to best select the rank-reducing transformation, or equivalently, the best AWVSS.

An important observation about this problem is that for a given stationary scenario and for a single point constraint in the GSC, there is no "best" multidimensional AWVSS, without additional constraints. That is, if one is free to choose any transformation, and one knows the scenario statistics *a priori*, then the best AWVSS is one-dimensional, and is given simply by the optimum full-rank weight vector. Any basis vectors that are orthogonal to the optimum full-rank weight vector may be used with it to create a multidimensional AWVSS, and all such AWVSS's will yield the same minimum array output power for that scenario. However, if the scenario statistics change, e.g. a jammer changes its waveform or a new jammer turns on, then the optimum weight vector will generally change. With an infinite number

of scenarios, the question becomes what multidimensional AWVSS will serve well for most of the expected scenarios.

Another use of the transformation is to statistically decorrelate the elements of the observation vector in the AWVSS. The well-known benefit of this decorrelation is fast convergence of the transform-domain normalized least mean squares (LMS) algorithm [16].

The filterbank [4] discussed in this report is a way of constraining the choices of rank-reducing transformations or AWVSS's to a small set with desirable properties. These properties include (1) that the filterbank has a tree structure that can adaptively grow as statistics are collected, (2) that there is a reasonable amount of decorrelation between the filterbank outputs, allowing for speedy convergence of the transform domain normalized LMS algorithm, and (3) that evaluating the transformation involves only addition and sign changes; it requires no floating point multiplications.

The results in this report include steady-state analysis of the performance of the reduced-rank GSC when a filterbank structure is used as the rank-reducing transformation. The array output power using the filterbank is compared to the un-transformed array output power with the same number of adaptive weights. We address the question of how to grow the tree structure and how to prune the tree to get reduced rank. We discovered that for an example linear array, there is a very significant preference for a particular filterbank structure, namely the wavelet structure. Furthermore, we show that for an example linear array, the penalty in array output power is very small if the wavelet structure is always used, regardless of the jamming scenario within a large collection of scenarios. As a further investigation of subspace preference, we developed a simple and straightforward monte carlo-type procedure to approximate the optimum AWVSS for a given collection of jamming parameters. We show the sensitivity to rank of the array output power using the monte carlo technique.

A MATLAB script was written to simulate the transform domain normalized LMS algorithm using filterbank transformations under nonstationary jamming conditions. The script allows the filterbank structure to change at regular intervals, in order to track the changing jamming conditions. A section of the report discusses some of the implementation details of this script, and includes some simulation results.

2 Notation and Geometrical Interpretation

The notation used in this report follows that of [5, 6]. The generalized sidelobe canceller, shown in Figure 1, is a way to implement a linearly-constrained tapped-delay-line (TDL) wideband adaptive array [2] so that an unconstrained adaptive algorithm may be used [2, 3]. We assume a uniform linear array and that a steering vector is used to ensure that the effective angle of the desired signal is broadside or 0 degrees from the normal to the array. For the minimum variance distortionless response (MVDR) array of K sensors and J taps per TDL, and a single constraint, the beamforming matrix takes the form [13]

$$\mathbf{W}_c = \frac{1}{K}\mathbf{1}$$

where $\mathbf{1}$ represents a column vector of K ones. The look direction signals are blocked from the adaptive processor by the $K-1 \times K$ signal blocking matrix, $\mathbf{W}_s$, such that $\mathbf{W}_s\mathbf{W}_c = 0$. For a five sensor array, for example,

$$\mathbf{W}_s = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}. \tag{1}$$

The data vector representing the entire array TDL structure is denoted by the KJ-dimensional column vector $X(k)$. $X(k)$ is made up of J snapshots of the array element outputs $x(k)$ such that

$$\mathbf{X}(k) = [\mathbf{x}^T(k)\ \mathbf{x}^T(k-1)\ \ldots\ \mathbf{x}^T(k-J+1)]^T$$

The $(K-1)J \times KJ$ block diagonal extended signal blocking matrix, $\mathbf{W}_{se}$, and the KJ-dimensional extended constraint vector, $\mathbf{W}_{ce}$, are given as

$$\mathbf{W}_{se} = \begin{bmatrix} \mathbf{W}_s & & & \\ & \mathbf{W}_s & & \\ & & \ddots & \\ & & & \mathbf{W}_s \end{bmatrix} \quad \mathbf{W}_{ce} = \begin{bmatrix} \mathbf{W}_c \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{0}$ denotes a vector with all-zero elements. The $(K-1)J$-dimensional noise space component is given by $\mathbf{X}_s(k) = \mathbf{W}_{se}\mathbf{X}(k)$ and the constraint

space component is given by $d(k) = \mathbf{W}_{ce}^H \mathbf{X}(k)$. The output of the array is

$$e(k) = d(k) - \mathbf{W}^H(k)\mathbf{X}_s(k),$$

where $\mathbf{W}(k)$ is the $(K-1)J$-dimensional adaptive weight vector. The optimal value of $\mathbf{W}(k)$ that minimizes the array output power solves the Wiener-Hopf equation $\mathbf{W}_{opt} = \mathbf{R}_{xs}^{-1}\mathbf{R}_{xd}$, where $\mathbf{R}_{xs} = E\{\mathbf{X}_s\mathbf{X}_s^H\}$, $\mathbf{R}_{xd} = E\{\mathbf{X}_s d\}$, and $\mathbf{X}_s^H$ is the conjugate transpose of $\mathbf{X}_s$. The average array output power, or the mean squared error, is given by $E\{e^2\}$. When the optimal weights are used, the minimum mean squared error (MMSE) is expressed

$$E\{e^2\} = r_{dd} - \mathbf{R}_{xd}^H \mathbf{R}_{xs}^{-1}\mathbf{R}_{xd},$$

where $r_{dd} = E\{d^2(k)\}$. When a transformation $\mathbf{T}$ is used on the noise space component, as in Figure 2, then the transform domain noise space component is $\mathbf{Z}(k) = \mathbf{T}\mathbf{X}_s$ and the output of the array is

$$e(k) = d(k) - \mathcal{W}^T(k)\mathbf{Z}(k).$$

The optimal weight vector in the transform domain is given by

$$\mathcal{W}_{opt} = \mathbf{R}_{zz}^{-1}\mathbf{R}_{zd}, \tag{2}$$

where $\mathbf{R}_{zz} = E\{\mathbf{Z}\mathbf{Z}^H\}$, $\mathbf{R}_{zd} = E\{\mathbf{Z}d\}$. The MMSE for the transform domain GSC is

$$E\{e^2\} = r_{dd} - \mathbf{R}_{zd}^H \mathbf{R}_{zz}^{-1}\mathbf{R}_{zd}.$$

If the transformation is non-singular, then the MMSE of the transform domain GSC and the standard GSC are equal. If the transformation is singular or rank deficient with rank M, then the MMSE of the transform domain GSC will generally be greater than for the standard GSC, and the increase in MMSE will depend on how far away the full-rank optimal weight vector $\mathbf{W}_{opt}$ is from the subspace spanned by the rows of $\mathbf{T}$. As in the case of the standard GSC, the performance of the transform domain GSC is invariant to nonsingular transformations following $\mathbf{T}$, so for steady state analysis, we may assume that $\mathbf{R}_{zz}$ is diagonal. In this case the MMSE becomes

$$E\{e^2\} = r_{dd} - \sum_{i=1}^{M} \frac{r_{zd,i}^2}{r_{zz,i}} \tag{3}$$

$$= r_{dd}\left(1 - \sum_{i=1}^{M} \rho_{zd,i}^2\right), \tag{4}$$

where $r_{zd,i}$ is the correlation of the ith element of $\mathbf{Z}(k)$, or $z_i(k)$, with the constraint space component $d(k)$, $r_{zz,i} = E\{z_i^2(k)\}$, and

$$\rho_{zd,i}^2 = \frac{E^2\{z_i(k)d(k)\}}{r_{zz,i}r_{dd}}.$$

$\rho_{zd,i}$ is the normalized correlation coefficient between $z_i(k)$ and $d(k)$; its magnitude is bounded by unity, and its nearness to unity indicates the quality of the linear homogeneous MMSE estimate of d by z_i. Let $\mathbf{R}_{xx} = E\{\mathbf{X}\mathbf{X}^H\}$ and $\mathbf{T}_i$ be the ith row of $\mathbf{T}$. Then $\rho_{zd,i} = \cos(\theta_i)$, where θ_i is the angle between [13, p. 194]

$$\mathbf{V} = \mathbf{R}_{xx}^{1/2}\mathbf{W}_{ce} \tag{5}$$

and

$$\mathbf{U}_i = \mathbf{R}_{xx}^{1/2}\mathbf{W}_{se}^H\mathbf{T}_i^H, \tag{6}$$

as illustrated in Figure 3. Note that when $\mathbf{R}_{zz}$ is diagonal, the $\mathbf{U}_i$'s are orthogonal vectors, but not necessarily normalized. One can casually think

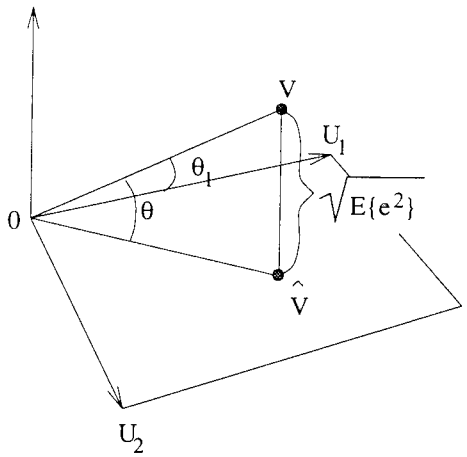


Figure 3: Geometrical interpretation of the transform domain GSC.

of $\rho_{zd,i}$ as the cosine of the angle between $\mathbf{W}_{ce}$ and $\mathbf{W}_{se}^H\mathbf{T}_i^H$, if it is implicit that the inner product is weighted with $\mathbf{R}_{xx}$.

Since the $\mathbf{U}_i$'s are orthogonal, we have that

$$\cos^2(\theta) = \sum_{i=1}^{M} \cos^2(\theta_i) = \sum_{i=1}^{M} \rho_{zd,i}^2, \tag{7}$$

where θ is the angle between $\mathbf{V}$ and the subspace spanned by the columns of $\mathbf{R}_{xx}^{1/2}\mathbf{W}_{se}^{H}\mathbf{T}$. Clearly, the closer that subspace is to $\mathbf{V}$, the lower is the MMSE of the transform domain GSC. $\hat{\mathbf{V}}$ is the projection of $\mathbf{V}$ onto the span of $\mathbf{R}_{xx}^{1/2}\mathbf{W}_{se}^{H}\mathbf{T}$, and therefore can be expressed

$$\hat{\mathbf{V}} = \sum_{i=1}^{M} = \frac{(\mathbf{V} \cdot \mathbf{U}_i)}{\|\mathbf{U}_i\|^2}\mathbf{U}_i, \tag{8}$$

where $(\cdot)$ denotes inner product. Substituting the expressions in equations (5) and (6) into equation (8) yields the dependence of $\hat{\mathbf{V}}$ on $\mathcal{W}_{opt}$ from in equation (2):

$$\hat{\mathbf{V}} = [\mathbf{U}_1\ \mathbf{U}_2\ \ldots\ \mathbf{U}_M]\,\mathcal{W}_{opt} \;=\; \mathbf{R}_{xx}^{1/2}\mathbf{W}_{se}^{H}\mathbf{T}^{H}\mathcal{W}_{opt}.$$

3 The Filterbank Transformation

The tree-structured filterbank used in this report is made up of quadrature mirror filter (QMF) blocks [4]. One QMF block is shown in Figure 4(a). The QMF block in this report is equal to the two-point DFT. The block consists of a high pass filter, represented by the z-transform $H(z) = 1 - z^{-1}$, and a low pass filter, represented by $L(z) = 1 + z^{-1}$, as shown in Figure 4(b). Each filter output is decimated-by-two. The filters create two subbands in the frequency domain, qualitatively shown on the right in Figure 4(b). The transitions could be made sharper by using longer filters (i.e. filters with longer impulse responses). The benefit of using longer filters would be improved decorrelation among the filterbank outputs, and hence faster convergence of the transform domain normalized LMS algorithm.

If an eight-element input sequence is given as $\mathbf{X} = [x_1,\ x_2,\ \ldots,\ x_8]^T$, then the decimated output of the high pass filter is $[x_2 - x_1,\ x_4 - x_3,\ x_6 - x_5,\ x_8 - x_7]^T$. The decimated output of the low pass filter is the same, except the minus signs are changed to plus signs. Thus the QMF block may

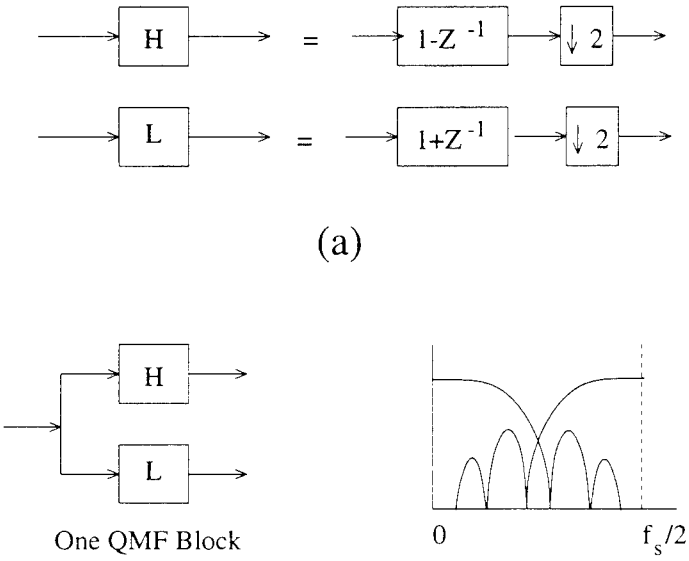


Figure 4: The quadrature mirror filter (QMF).

be represented by a matrix **T1**:

$$\mathbf{T1} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

and the output of the block is $\mathbf{T1} * \mathbf{X}_s$.

The QMF blocks are arranged in a tree structure to create the filterbank. Several example tree structures are shown in Figure 5. These structures will be refered to, in the remainder of this report, by TLL, TLH, THL, THH, and TU, respectively. The TLL structure in Figure 5(a) can be described by **T1** multiplied by the matrices described below.

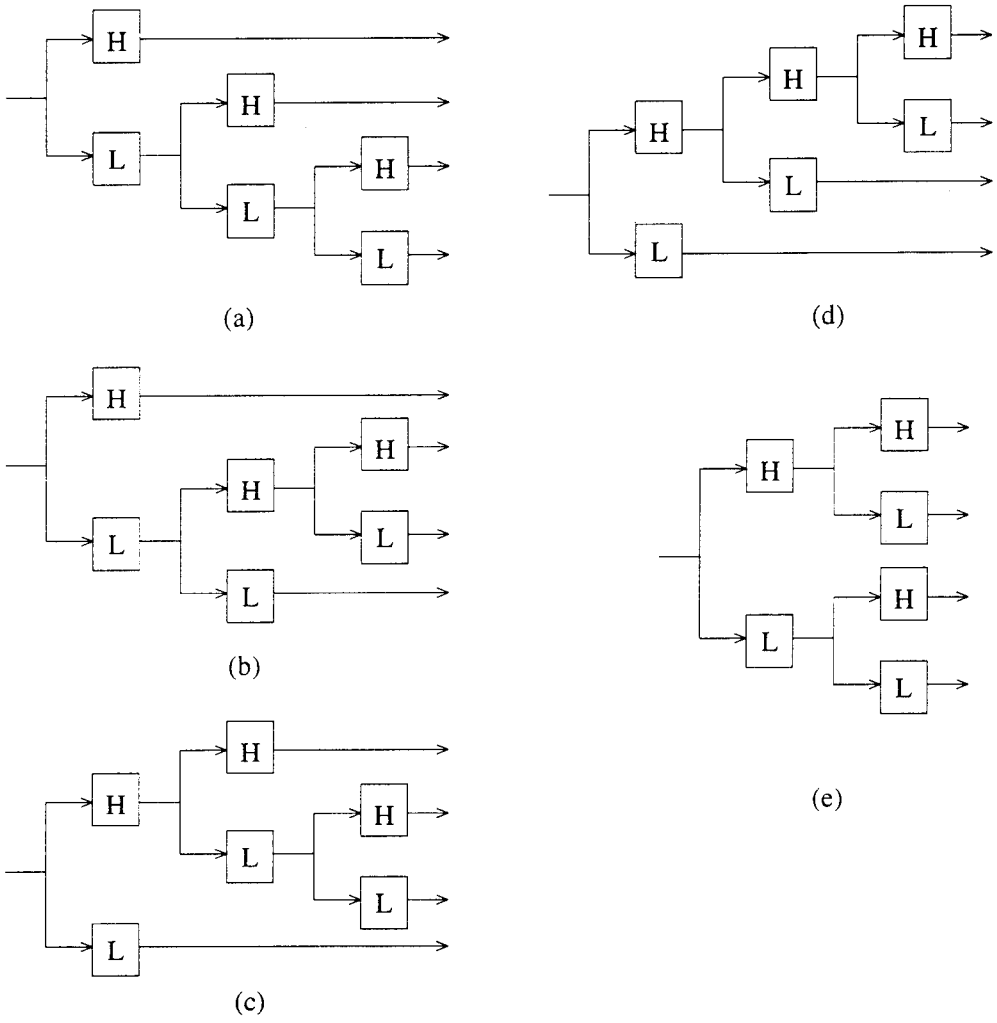


Figure 5: The five filterbank structures available for the five sensor, two tap, single constraint case.

The second tier of the filterbank in Figure 5(a) can be represented by another square matrix **T2**:

$$\mathbf{T2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix},$$

and the third tier by **T3**:

$$\mathbf{T3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

The entire filterbank may be represented by a matrix **T**, that is the product of the above three matrices $\mathbf{T} = \mathbf{T3} * \mathbf{T2} * \mathbf{T1}$, and

$$\mathbf{T} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ -1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

The widest band has four output samples, the next widest band has two output samples, and the narrowest band has only one output sample. Regardless of how the tree is grown, there will always be the same number of total output samples as there are input samples. Thus **T** is square and furthermore

it is not singular; therefore using it has no effect on the MMSE of the GSC and it does not reduce the number of adaptive weights. To achieve rank reduction, some rows of $\mathbf{T}$ must be pruned away. The result of pruning will be a non-square, $M \times (K-1)J$ matrix, where $M < (K-1)J$. The challenge is to grow the right tree structure and then prune intelligently. In [5, 6], an eight input example like this one was used and only the first sample was kept from each band, resulting in a rank reduction of four. As we will show in a later section, the choice of which sample to keep can make a difference of up to 3 dB in MMSE.

3.1 Growing the Filterbank

We have investigated several filterbank growth algorithms. In our early filterbank growth algorithm [6], the filterbank was initialized with just $\mathbf{T1}$. The energy in each of the two bands was estimated and then the band with the highest energy was further refined with another QMF block, yielding three bands. Next, the band with the highest energy of the three was further refined, and so on. In other words, the filterbank grew in the direction of the most energy. The filterbank attempted to equalize the power in its bands, placing the narrowest bands in the regions of highest energy. Once the filterbank structure was complete, the first sample of each band was kept and the rest were pruned away. We then realized that this growth and pruning method totally ignored the correlation between the filterbank outputs and the constraint space component. As shown in the previous section, choosing a transformation that maximizes these correlation coefficients in turn minimizes the MMSE. For example, jammer A could have a direction of arrival that coincides with a null in the quiescent pattern or constraint space component, and have significant energy relative to the other jammers in the auxiliary beams or noise space component. The energy-based growth algorithm will grow the filterbank in the directions of jammer A's energy, and prune away directions that relate to the other jammers, and the result would lead to poor correlation coefficients.

A later filterbank growth algorithm, reported in [6], estimated the sum in equation (3) for each of a finite collection of pruned filterbank structures. A simulation was run for 3 sensors, one directional constraint, and 4 taps on each $\mathbf{W}_s$ so that $\mathbf{X}_s$ was eight-dimensional. A tree-structured filterbank, using the two-point DFT QMF blocks, was used to reduce the number of

Signal	DOA	SNR in dB	f_c	% BW
desired	0^o	0	0.10	12 %
jammer 1	45^o	17	0.16	2 %
jammer 2	60^o	32	0.08	1 %

Table 1: Signal parameters for the filterbank simulation

adaptive weights to four. The five filterbank structures that use three half-band refinements are shown in Figure 5. The first sample from each band was kept and the others were pruned away. There were two jammers and one desired signal illuminating the array. The jammer and signal parameters are given in Table 1.

The steady-state array patterns using the optimal weights are presented for the four-weight TDL processor, the four-weight filterbank processor, and the eight-weight TDL processor in Figures 6, 7, and 8, respectively. The four-weight filterbank pattern matches very closely the eight-weight TDL pattern, and the MMSE is very close as well. The MMSE for the four-weight TDL was 13.5 dB, for the four-weight filterbank was 10.5 dB, and for the eight-weight TDL was 10.03 dB. The filterbank therefore gave a 3 dB improvement over the TDL processor with the same number of adaptive weights. The transform-domain normalized LMS algorithm was simulated MMSE-based growth algorithm. The learning curves for the four-weight TDL processor and the four-tap filterbank processor, averaged over 1000 independent simulations, are shown in Figure 9. It appears that the filterbank improves convergence time over the four-weight TDL in addition to lowering the MMSE.

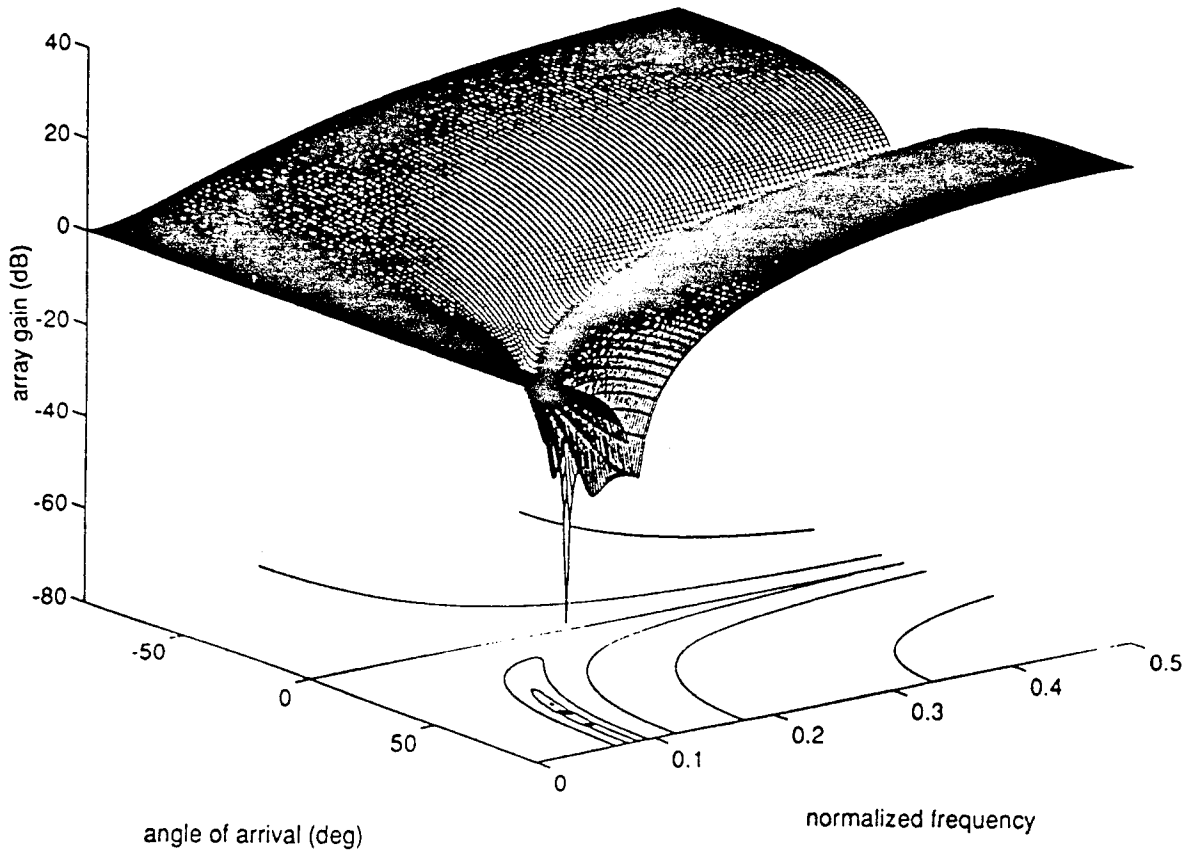


Figure 6: Antenna gain pattern for the four-weight TDL processor. The MMSE is 13.5 dB.

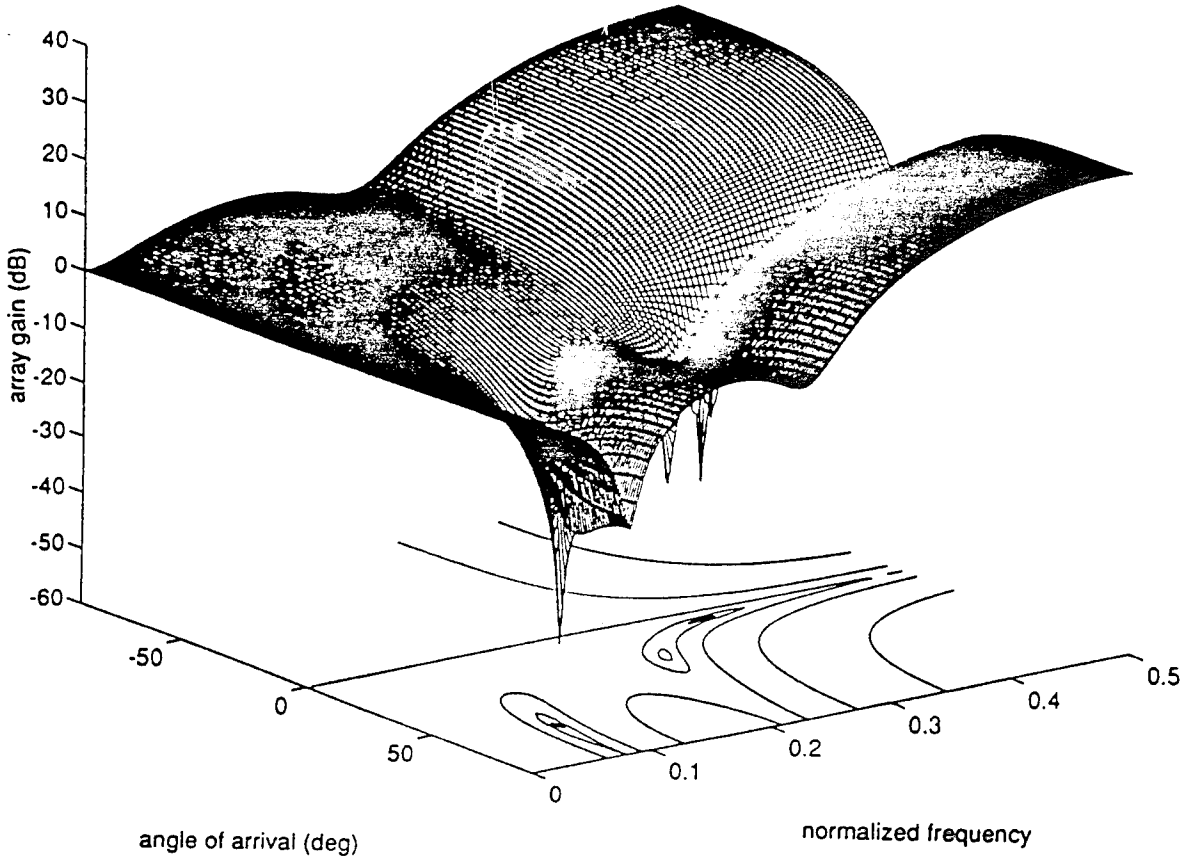


Figure 7: Antenna gain pattern for the four-weight filterbank processor. The MMSE is 10.5 dB.

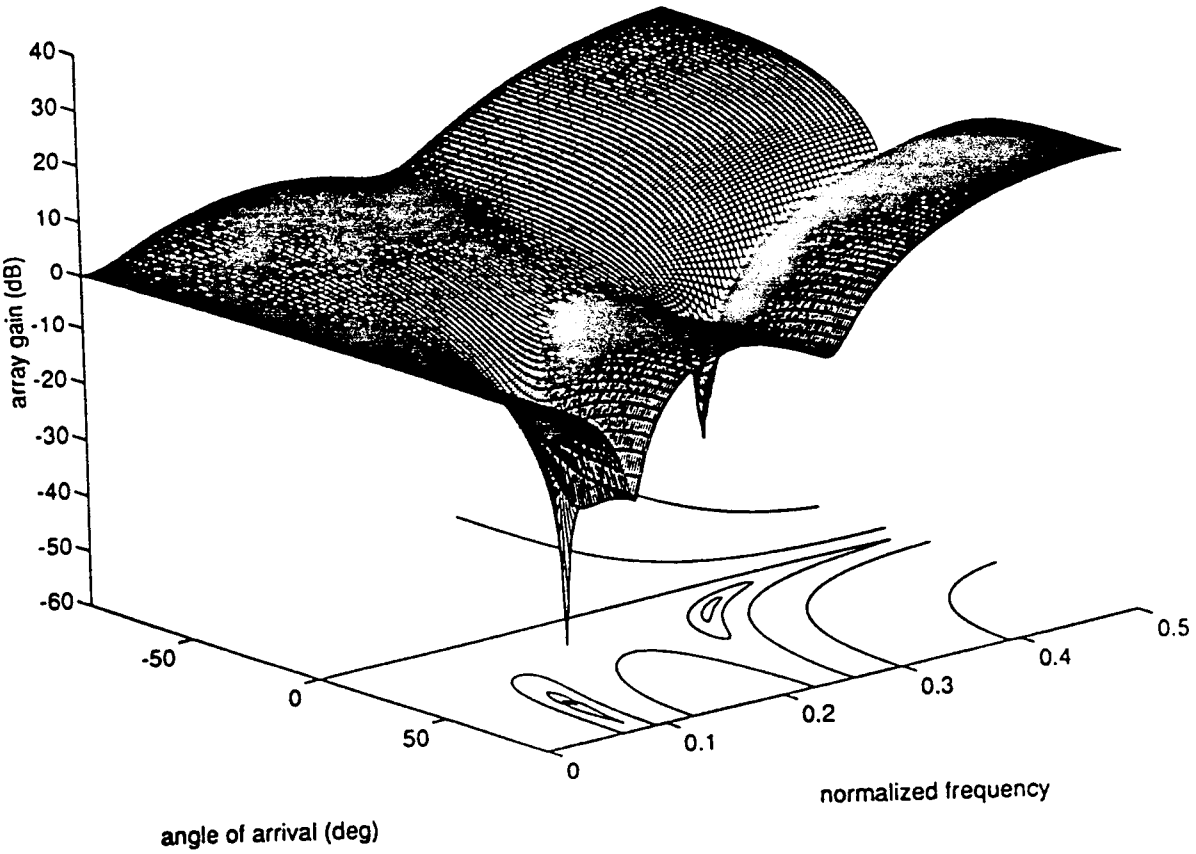


Figure 8: Antenna gain pattern for the eight-weight TDL processor. The MMSE is 10.03 dB.

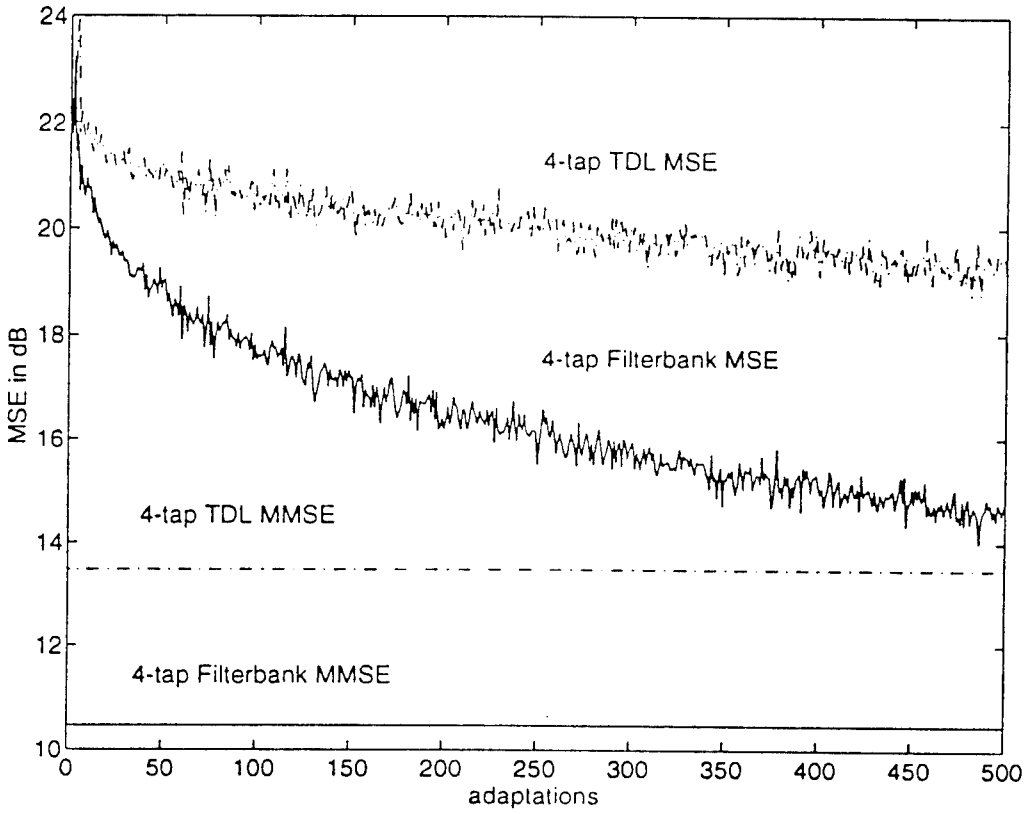


Figure 9: Learning curves for the four-weight TDL processor and the four-weight filterbank processor, averaged over 1000 trials.

A variation on the growth algorithm in [6] is to simply refine the band that contains the maximum absolute value of $\rho zd, i$ over i. If the transformation $\mathbf{T}$ produces a diagonal $\mathbf{R}_{zz}$, then this method is suboptimal since the angle θ can be small even though θ_i is not small for any i (see equation (7)). However, if $\mathbf{R}_{zz}$ is not diagonal, then this method gives results comparable to that of [6], and is alittle easier to program.

3.2 Pruning Choices

In an earlier section, we observed that simply growing a filterbank does not provide rank reduction. The tree must be pruned. Since refinements are performed on branches that have high correlation with the constraint space component, we look to the branches that are not refined for pruning. The branches that are not refined have several samples at their outputs for every one sample of the longest filter (or branch). For example, in the TLL structure of Figure 5(a), there are four samples at the output of the uppermost branch and two samples at the next uppermost branch. If we decide to keep one sample from each band (for decorrelation purposes), then we have $4 \times 2 = 8$ choices for pairs of samples to keep. We wondered if there was much difference between these eight pruning choices in terms of MMSE of the rank-reduced array. We found that there is often a significant difference in MMSE between these choices. An experiment was conducted using 500 randomly selected jamming scenarios using the 5 sensor, 2 tap MVDR array. Note that the $\mathbf{W}_s$ matrix for this experiment is different than the one in equation (1), but has the same span. We allowed the parameters for three barrage noise jammers to be independently and uniformaly distributed over the following ranges: [-90°,+90°] in angle of arrival, [0, 30] in jammer-to-noise (JNR) ratio in dB, [0,1] in normalized center frequency, and [0,30] in percent bandwidth. For each random scenario or trial, the MMSE was computed for each pruning choice for each filterbank structure. For each structure, the minimum MMSE choice and the maximum MMSE choice was identified, and the difference between maximum and minimum MMSE was recorded. The trials were grouped according to which filterbank structure had the lowest MMSE (using the best pruning choice). Figure 10 contains histograms of the differences between min and max MMSE in dB. The upper left histogram corresponds to the group of 319 trials for which the TLL structure was the best. We observe that for the majority of trials, at least 250 of them, there

Signal	DOA	SNR in dB	f_c	% BW
desired	0^o	0	0.89	0.9 %
jammer 1	27.2^o	24.4	0.76	24 %
jammer 2	-34.5^o	26.9	0.17	16 %
jammer 3	-79.1^o	27.5	0.57	26 %

Table 2: Scenario for which best structure was TLL and for which the pruning choice MMSE difference was 1.09 dB.

Signal	DOA	SNR in dB	f_c	% BW
desired	0^o	0	0.27	13 %
jammer 1	4.39^o	13.2	0.95	12 %
jammer 2	-17.7^o	22.0	0.74	25 %
jammer 3	15.6^o	22.1	0.68	1 %

Table 3: Scenario for which best structure was TLL and for which the pruning choice MMSE difference was 3.4 dB.

is less than one dB difference in MMSE between the eight pruning choices. However, the differences were as much as 3 dB in a few trials. The large difference cases do not necessarily correspond to stronger jammers than the 1 dB difference cases. For example, a scenario with the parameters in Table 2 produced a pruning choice difference of 1.09 dB, while the scenario with the parameters in Table 3 produced the pruning choice difference of 3.4 dB.

The remaining histograms indicate that when the optimal filterbank structure is not TLL, then the pruning choice can make a difference of 4 dB or higher.

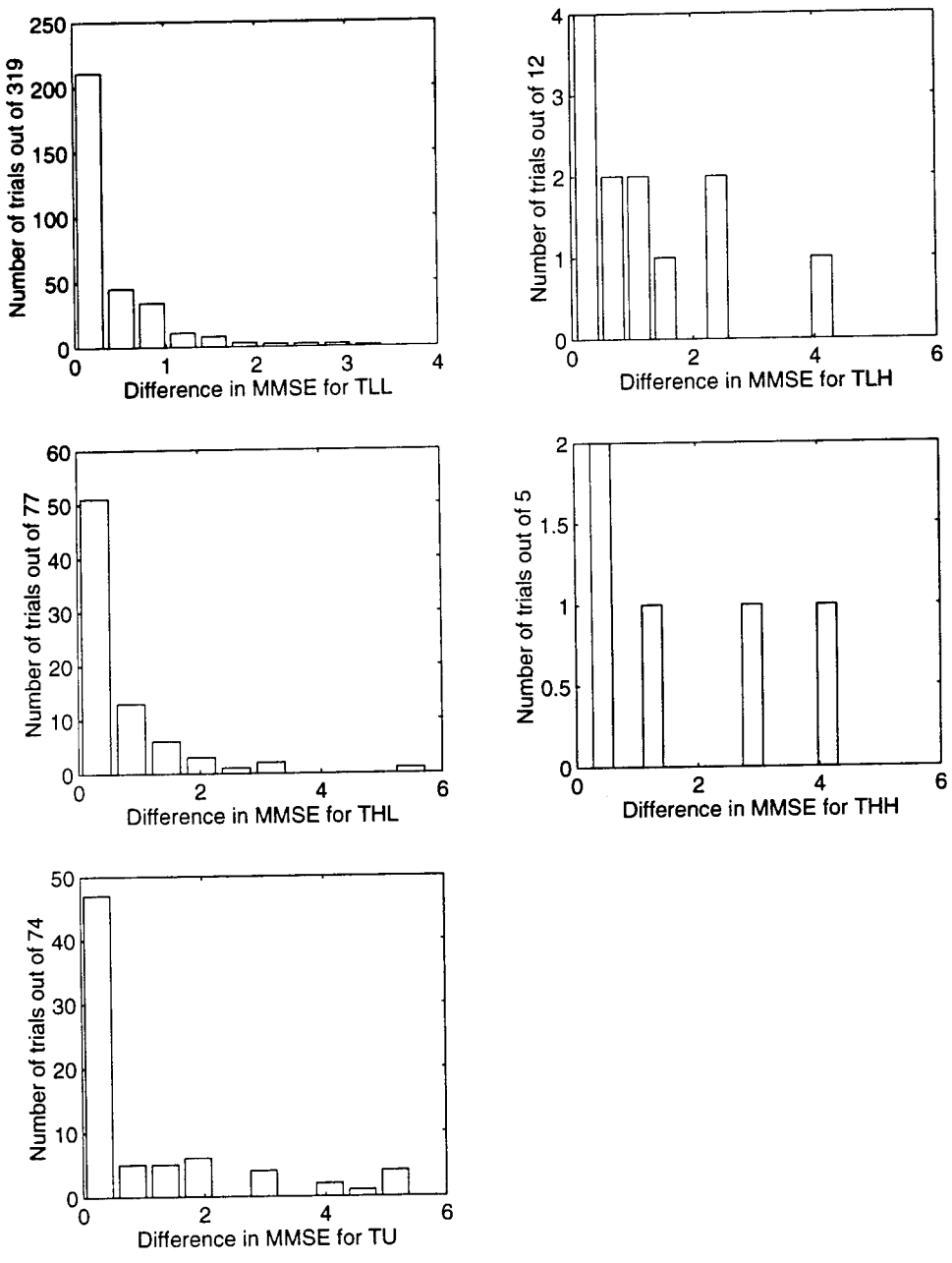


Figure 10: Difference in dB between maximum and minimum MMSE decimation choices for each filterbank structure.

	Filterbank Structure				
	TLL	TLH	THL	THH	TU
Number of trials	1655	3	214	4	124

Table 4: Statistics for filterbank structures that yield the lowest MMSE for a set of 2000 random scenarios.

4 Subspace Selection Using Prior Statistics

We found that we had to conduct an extensive search to find scenarios for which the minimum MMSE filterbank structures were different. This led us to investigate the general question of what is the best AWVSS. To illustrate the filterbank issue, we present the following example of the five sensor uniform linear LCMV array with two taps and one directional constraint. We allowed the parameters for three barrage noise jammers to be uniformaly distributed over the following ranges: [-90°,+90°] in angle of arrival, [-20, +20] in jammer-to-noise (JNR) ratio in dB, [0,0.5] in normalized center frequency, and [0.30] in percent bandwidth. Observe that the JNR range implies that the scenario can have effectively one, two, or three jammers. For each random trial of the scenario parameters, we determined the best structure and the best pruning of that structure in terms of MMSE, always keeping the samples from the longest two filters. If the best structure was not structure (a), then we computed the penalty in MMSE that would result if structure (a) were used. The results of this analysis are shown in Table 2. We found that 80% of the time filterbank structure (a) was preferred. Furthermore, for the cases that another filterbank structure minimized output power, the penalty for using structure (a) was slight: < 0.5 dB for 93% of the cases, < 1 dB in 98% and < 2.5 dB for the remainder of cases. This means that algorithms that involve time-varying filterbank structures need to be carefully evaluated in terms of the trade-off between cost of implementation and expected improvement over a fixed filterbank or other fixed transformation.

After making the observation that changing filterbank structures is often not necessary, we considered the question of what is the best M-dimensional subspace for the adaptive weights. Several other authors [8, 9, 10, 11, 12, 13, 14] have considered this question.

Van Veen's method [9, 10, 11, 13] is to minimize the average output power of the LCMV array, where the average is taken over a selection of interference scenarios, and the minimization is over all transformations of a specified rank. In [12], he considers optimizing over the quiescent pattern in addition to the rank-reducing transformation. Ma and Griffiths [8] approach was to compute a set of sample full-rank optimum weight vectors in the noise space and then determine the solution hyperplane that minimizes the sum of the squared distances of the full rank optimum weight vectors to the solution subspace. Van Veen's approach is optimal in the sense that he is optimizing over the array output power, but the calculation appears to be rather difficult. The approach in [8], while suboptimal in terms of array output power, involves a calculation that is less complex. The collection of scenarios in both [8] and [12] are limited to variations in interferer directions. These papers all deal with beamformer design using *a priori* statistics and not with real-time calculation, although the techniques are probably amenable to real-time implementation. The authors in [14], on the other hand, consider a very interesting real-time calculation that decides first how many adaptive weights there should be based on the Akaike Information Criterion and the Minimum Description Length criterion, and second, the appropriate space for those adaptive weights using a Gram-Schmidt orthogonalization procedure that incorporates the constraint space component.

The method reported here takes an approach very similar to that used in [8], and considers a much larger variation in scenarios. The details of our method are described next and then the method is compared to [8].

For each random selection of jamming and signal parameters described in the opening paragraph of this section, the optimal full-rank weight vector, $\mathbf{W}_{opt}$, is computed. These column vectors are collected into a matrix $A = [\mathbf{W}_{opt(1)}, \mathbf{W}_{opt(2)}, \ldots, \mathbf{W}_{opt(N)}]$, where $\mathbf{W}_{opt(i)}$ is the optimal full rank vector for the ith random selection. The singular value decomposition (SVD) is performed on A^T, $A^T = USV^H$. If a rank M transformation is desired, then the $\mathbf{T}^T$ is designated to be the first M columns of V. This solution optimizes the matrix norm of the difference between A and $\mathbf{T}^T$ [15].

In [8], a mean weight vector is first computed, $\sum_{i=1}^{N} \frac{1}{N} \mathbf{W}_{opt(i)}$, and then the mean vector is subtracted from each optimal weight vector prior to the formation of A. Then an eigendecomposition is performed on AA^H. The eigenvectors of AA^H corresponding to the $(K-1)J-M$ smallest eigenvalues are used to augement the constraint on the adaptive weight vector.

We now present a numerical example. We calculated the full-rank optimal weight vector for each of 1000 random trials. Next the SVD method was used to identify the best rank M transformation, where M ranges from 1 to $(K-1)J$ for the five sensor, two tap example. We then ran an additional 1000 trials, where in each trial we calculated the difference in MMSE for the full-rank optimal weight vector and the M-dimensional optimal weight vector, or the MMSE penalty, using the rank-M transformation calculated from the previous 1000 trials. We did this for two different wideband arrays. The first example array is the 5 sensor, 2 tap MVDR array used in previous examples. The second array is the 11 sensor, 8 tap MVDR array. Figures 11 and 12 correspond to the first example and Figures 13 and 14 correspond to the second array. Each figure contains several curves. Consider the curve in Figure 11, labelled 90%, for example. For a rank value of two, this curve takes the value 1.5. This means that in 90% of the 1000 trials, use of the rank-two transformation incurred an MMSE penalty less than 1.5. Figure 12 has the same curves as Figure 11, plus the 100% curve. If we look at the rank 4 transformation, we see that 95% of the time the penalty is negligible, and all of the penalties in these 1000 trials were less than about 1.5 dB. If we use the rank 5 transformation, all penalties are negligible. To check that this phenomenon was not peculiar to the 5 sensor, 2 tap case, we tried 11 sensors and 8 taps. The rank 4 transformation is still pretty good, incurring penalties less than 2 dB 95% of the time. The rank 12 transformation was required to ensure that all penalties were negligible. It is remarkable that with no rank reduction, the second array has 80 degrees of freedom; yet a fixed transformation taking the number of adaptive weights from 80 down to 12 incurs no significant penalty over 1000 random scenarios. These results suggest that the appropriate rank-reducing transformation may not depend as much on the scenario as originally thought.

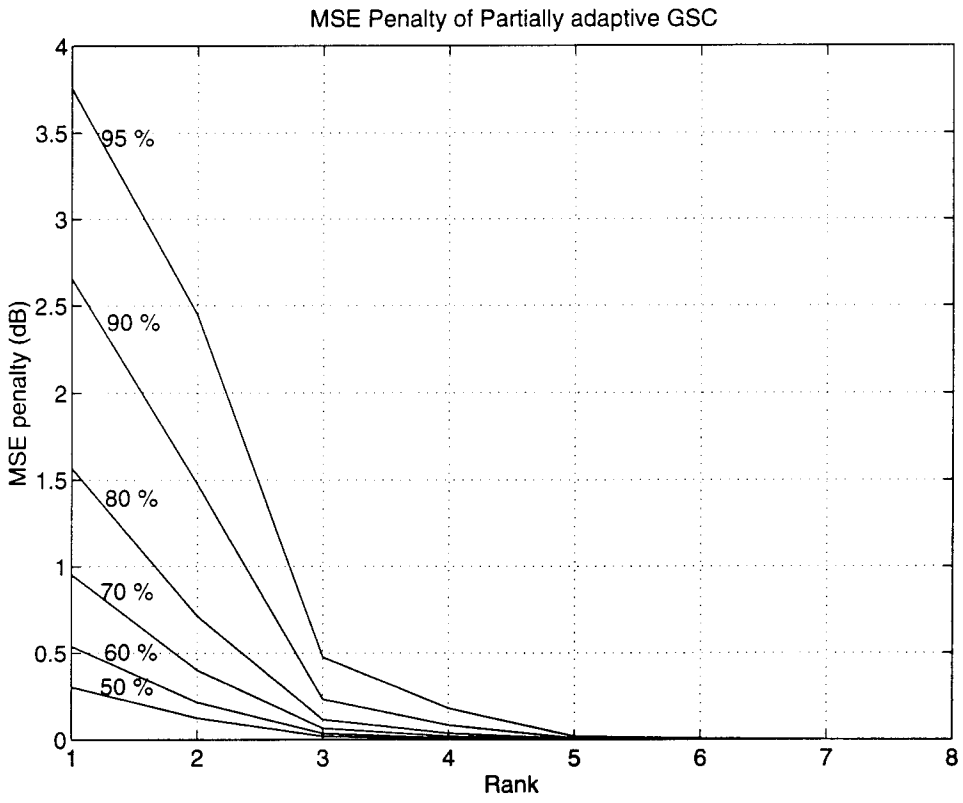


Figure 11: Averaged MSE Penalty of partially adaptive GSC for 1000 trials

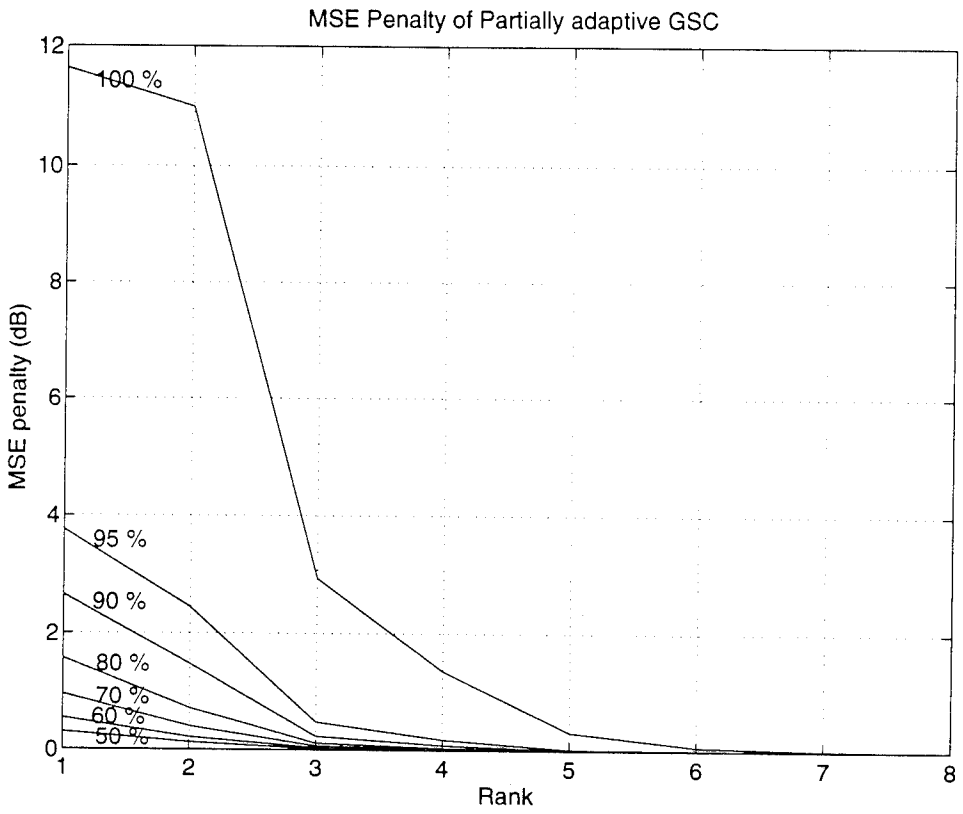


Figure 12: Averaged MSE Penalty of partially adaptive GSC for 1000 trials

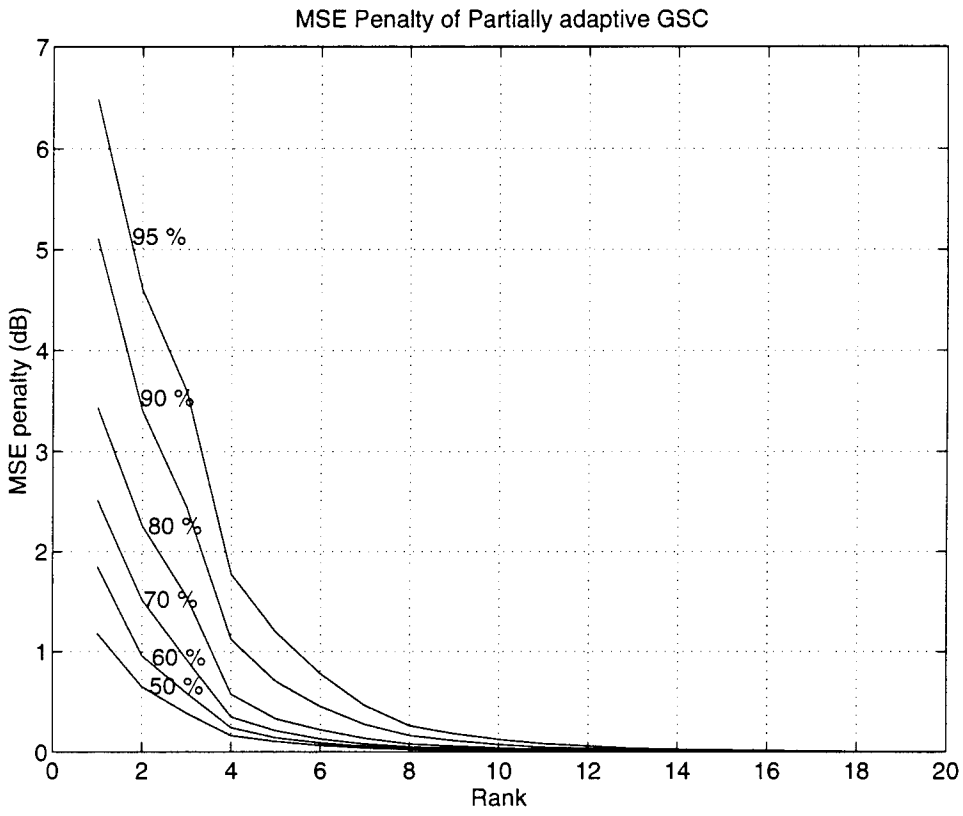


Figure 13: Averaged MSE Penalty of partially adaptive GSC for 1000 trials

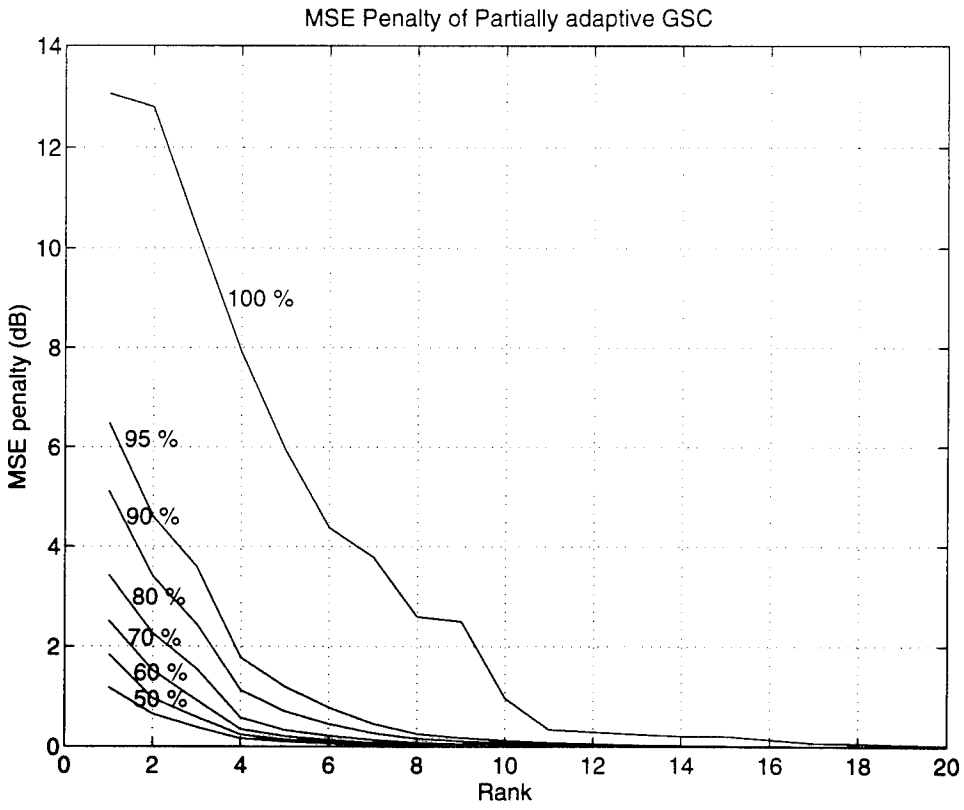


Figure 14: Averaged MSE Penalty of partially adaptive GSC for 1000 trials

5 The Time-Varying Filterbank

Suppose that a certain jamming scenario has a particular minimum MMSE filterbank structure, but then a jammer turns off, moves, changes in some way or another jammer turns on. The new scenario may have a different minimum MMSE filterbank scenario. An adaptive algorithm could be made to sense the need to change the filterbank, if the minimum MSEE values for the two structures in the second scenario are different enough. A MATLAB script was written to simulate such an algorithm along with the normalized transform-domain LMS algorithm for the five sensor, two tap, single directional constraint MVDR array. This section discusses some of the implementation issues and presents some example simulation results.

5.1 Implementation Notes

Before description of the time-varying filterbank (TVFB) algorithm, it will be convenient to define some additional notation. As before, $\mathbf{T}$ is the square matrix that represents the unpruned filterbank and $\mathbf{Z}$ is the output of the unpruned tree, so $\mathbf{Z} = \mathbf{T} * \mathbf{X}_s$. Let $\tilde{\mathbf{T}}$ be the nonsquare matrix that represents the pruned filterbank; $\tilde{\mathbf{T}}$ is made up of selected rows of $\mathbf{T}$. Let $\tilde{\mathbf{Z}} = \tilde{\mathbf{T}} * \mathbf{X_s}$ be the output of the pruned tree.

Using single-pole filters, the algorithm estimates $\mathbf{R}_{zz}$, the autocorrelation matrix for $\mathbf{Z}$, and $\mathbf{R}_{zd}$, the crosscorrelation of $\mathbf{Z}$ with $d(k)$. It may be possible to get by with estimating $\mathrm{diag}(\mathbf{R}_{zz})$ instead of the full matrix $\mathbf{R}_{zz}$, but we found the the full matrix is needed to properly regrow the filterbank when the need is detected. The normalized correlation coefficients $\rho_{zd,i}$ are computed using the smoothed estimates of $\mathrm{diag}(\mathbf{R}_{zz})$ and $\mathbf{R}_{zd}$ at each timestep.

The filterbank is initialized to the most preferred structure (TLL for the example array simulated) and the pruning choice was arbitrarily chosen to be the first sample in each band. At regular intervals, the algorithm checks to see if the filterbank should be changed. The detection criterion is based on the variation on the growth algorithm in [6] discussed at the end of Section 3.1. The detection process is simply to see if the maximum of $\rho_{zd,i}$ over i corresponds to one of the longest filters. This test is based on the assumption that since the filterbank is grown in the direction of maximum correlation coefficient, that the output of the longest filter will have the highest normalized correlation coefficient. In the event that a filterbank output from other than

the longest filter had the largest correlation coefficient, then the statistics for the output of the first filterbank stage are obtained from those of the present fully grown filterbank. These statistics are the autocorrelation matrix

$$(\mathbf{T3} * \mathbf{T2})^{-1}\mathbf{R_{zz}}(\mathbf{T3} * \mathbf{T2})^{-\mathbf{T}} \tag{9}$$

and the crosscorrelation $(\mathbf{T3} * \mathbf{T2})^{-1}\mathbf{R_{zd}}$. The normalized correlation coefficients for the first stage are then estimated, and the filterbank is regrown in the direction of largest correlation coefficient. The need for having an estimate of the full autocorrelation matrix $\mathbf{R}_{zz}$ is evident from equation (9). After the filterbank is grown to have the longest possible filter, then pruning takes place. The longest filter outputs are kept and then from the remaining outputs, the outputs with the largest correlation coefficients are kept until the desired rank is attained.

When the filterbank changes structure, the MSE can be kept low by properly mapping the LMS weights and re-setting diag($\mathbf{R}_{zz}$) to keep the proper LMS step size. Noting that the rows of $\mathbf{DT}$ are normalized and orthogonal, the weight mapping is:

$$\mathbf{W}_{fb,\text{new}} = \tilde{\mathbf{T}}_{\text{new}} * \tilde{\mathbf{T}}^{T}_{\text{old}} * \mathbf{W}_{fb,\text{old}}$$

This mapping ensures that weights that correspond to $\mathbf{U}_i$'s that are shared by the old and new $\mathbf{DT}$'s will be preserved (perhaps shuffled), and not set to zero.

We found it helpful to reinitialize the estimates of $\mathbf{R}_{zz}$ and $\mathbf{R}_{zd}$ after each filterbank check. Although hundreds of timesteps are needed for the estimates of $\mathbf{R}_{zz}$ and $\mathbf{R}_{zd}$ to converge to their true values, we found that not many steps were needed (20 or less) for the estimates of $\rho_{zd,i}$ to become distinct enough for accurate filterbank growth.

5.2 Simulation Results

In this section we present simulation results for the 5 sensor, 2 tap, single directional constrain MVDR array, that has been used throughout this report. We simulated for 700 timesteps. The scenario described in Table 5 lasted from the beginning up to timestep 200, at which time the scenario changed to that of Table 6. These scenarios were picked because in a collection of

500 randomly generated scenarios, they prefered structures (a) and (b), respectively, and they had the largest difference in minimum MMSE from any of the other filterbank structures among other scenarios that prefered their same respective structures. Specifically, the first scenario has a steady state minimum MMSE of 8.12 dB when the TLL structure is used and any other structure yields a minimum MMSE of at least 10.04 dB. The second scenario has a steady state minimum MMSE of 4.05 dB for the TLH structure, and any other structure yields a minimum MMSE of at least 6.37 dB. In particular, the minimum MMSE for the second scenario and the TLL structure is 7.19 dB, a 3.14 dB difference from TLH. Since preferred filterbank structures other than TLL are hard to find, we have provided the steady state patterns for second scenario and the optimum filterbank structure. Figures 15 through 20 show three sets of pattern cuts and three complete patterns. The first two of these figures correspond to the full rank optimal weight vector (8 dimensional) for 5 sensors, 2 taps. The second two of these figures correspond to the full rank optimal weight vector (4 dimensional) for 3 sensors and 2 taps. And the last two of these figures correspond to the reduced rank optimal weight vector (4 dimensional) for 5 sensors, 2 taps, using the TLH filterbank with optimal pruning. Each pattern cut is for constant frequency and variable angle to ensure that nulls are produced in the right places. Each plot has three curves correponding to the exact frequency of a jammer and plus or minus 0.01 in normalized frequency. We note that in these simulations, the antenna elements are $\lambda/2$ apart for the highest frequency of unity and the taps are spaced at the Nyquist rate for the highest frequency. One thing to notice about the complete patterns is that a wide null is required for the powerful second jammer. The 8-dimensional Tapped Delay Line (TDL) and the 4-dimensional filterbank arrays can put down a wide null. However, the 4-dimensional TDL array cannot do as well.

The algorithm checked for currect filterbank structure every 150 timesteps, starting at 150. The sequence of events is thus: at 150, the algorithm has a chance to change the filterbank structure from its initial TLL to something else. In this simulation it stayed with TLL but changed its pruning choice. At 200, the scenario changed, while the filterbank was held at TLL. At 300, the algorithm had another opportunity to change structures, and this time it chose TLH, which is the optimal structure for this scenario, and therefore a correct choice. At 450 and 600, there were other opportunities for structure change but the structure remained at TLH. Figures 21 and 22 show how

the MSE and the real values of the adaptive weights evolve over time. This MSE is not smoothed; it is simply the squared value of the array output. The LMS algorithm appears to be converging prior to change in pruning choice at 150. The MSE increases somewhat after the pruning change, because of the subspace change. The initial convergence is also evident in the weight trajectories. The pruning change is evident also as one of the weight real parts goes to zero at 150. This is because the dimension (i.e. the $\mathbf{U}_i$) corresponding to that adaptive weight in the initial pruning choice does not exist in the space spanned by the second pruning choice at 150. The weights appear to be converging again between 150 and 200, and presumably they would converge and provide a lower MSE because of the better pruning choice. But they do not converge before the scenario changes at 200, creating a big spike (+500) in MSE. The weights again appear to be converging but the MSE has grown because the filterbank has the wrong structure for the new scenario. At 300, the algorithm changes the structure to the correct one, the weights converge and the MSE settles down. We reran this set of scenarios several times using different random seeds, with similar results.

6 Conclusions

This report has explored various issues associated with time-varying partially-adaptive arrays using the generalized sidelobe canceller framework, and the use of a simple filterbank structure to achieve rank reduction. We observed that many reasonable jamming scenarios can be well accomodated with a fixed rank-reducing transformation. The tree-structured filterbank offers a finite collection of rank-reducing transformations with desirable qualities. The filterbank can be implemented using only addition and subtraction; no multiplication. The longer filters of the filterbank, when it is grown in the directions of best normalized correlation coefficient, span a subspace for the adaptive weights which is close in angle to the signal to be cancelled. The longer filters provide some decorrelation between inputs to the adaptive weights, therefore yielding fast convergence of the transform-domain normalized LMS algorithm. Pruning of the shorter filters achieves rank reduction with a small penalty in MMSE relative the the optimal full-rank MMSE.

Signal	DOA	SNR in dB	f_c	% BW
desired	0^o	0	0.70	21 %
jammer 1	7.30^o	18.96	0.51	28 %
jammer 2	-69.1^o	16.49	0.72	11 %
jammer 3	41.9^o	15.32	0.97	19 %

Table 5: The first scenario in the simulation. This scenario prefers the TLL or structure (a).

Signal	DOA	SNR in dB	f_c	% BW
desired	0^o	0	0.07	5.3 %
jammer 1	30.6^o	17.98	0.58	24.4 %
jammer 2	-69.2^o	27.94	0.36	7.9 %
jammer 3	43.15^o	6.42	0.96	27.1 %

Table 6: The second scenario in the simulation. This scenario prefers the TLH or structure (b).

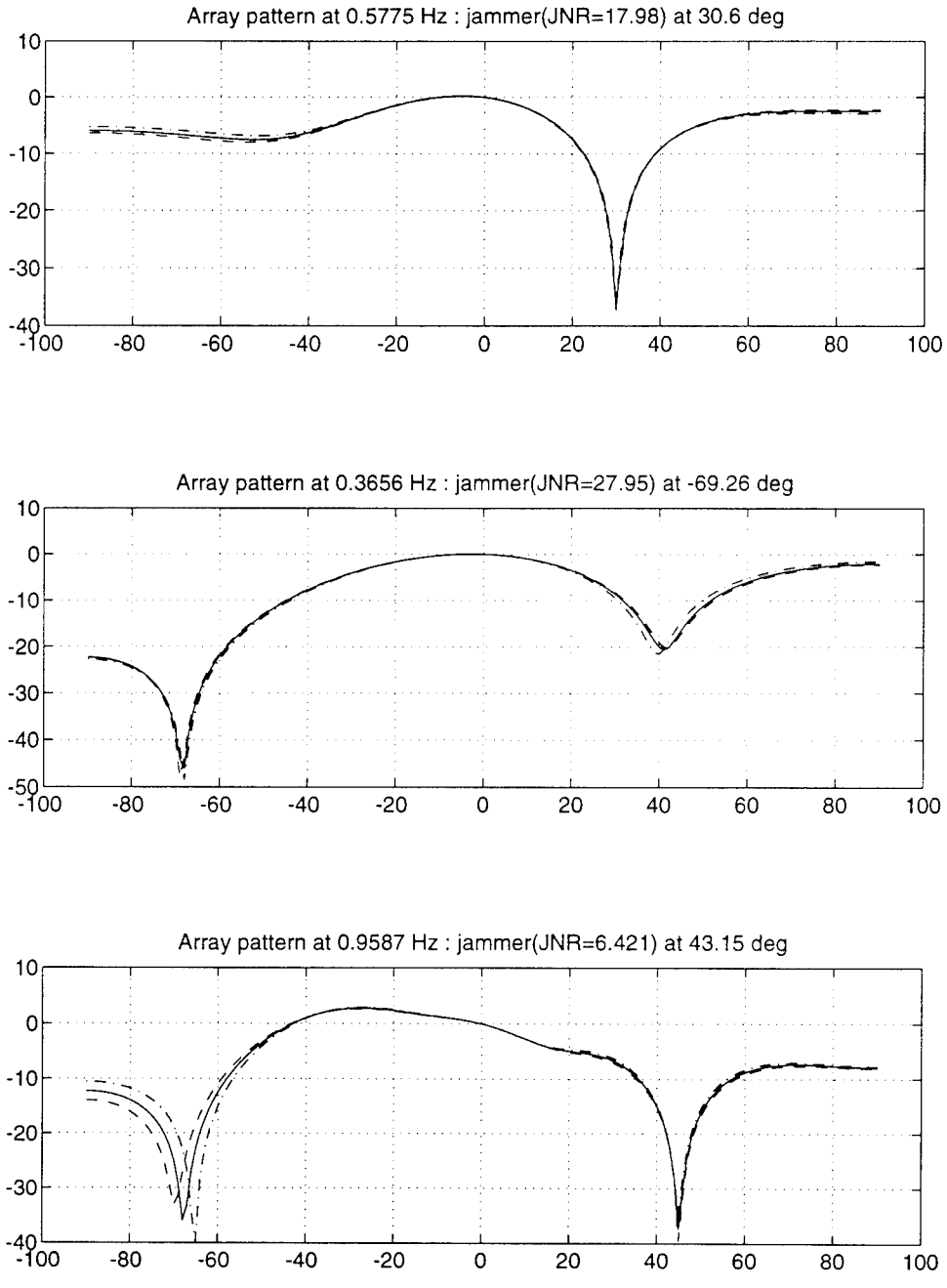


Figure 15: Pattern cuts for 5 sensor, 2 tap full rank optimal solution

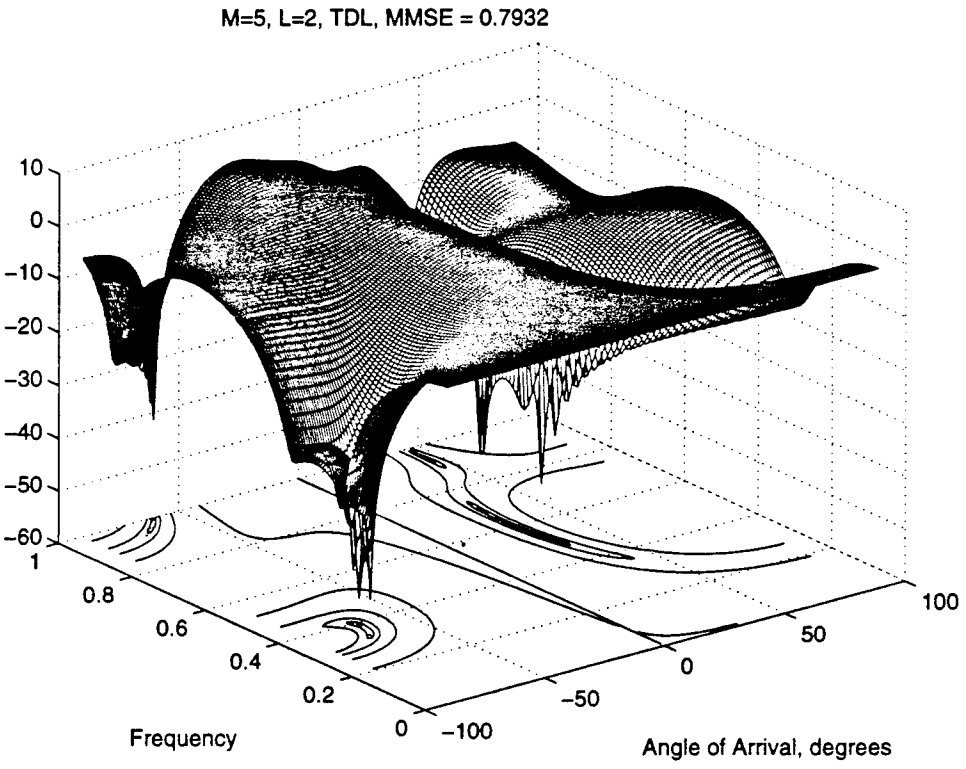
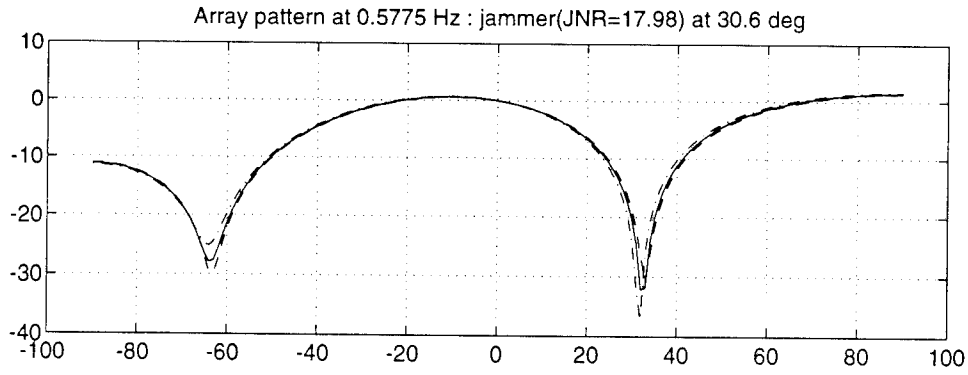
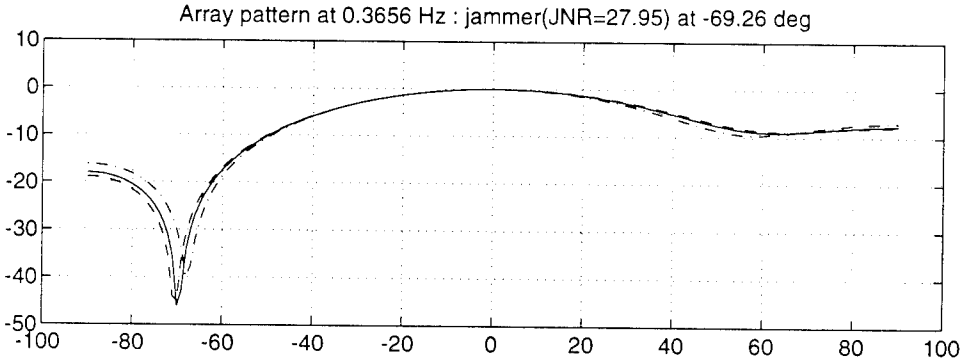


Figure 16: Complete pattern for full rank optimal solution





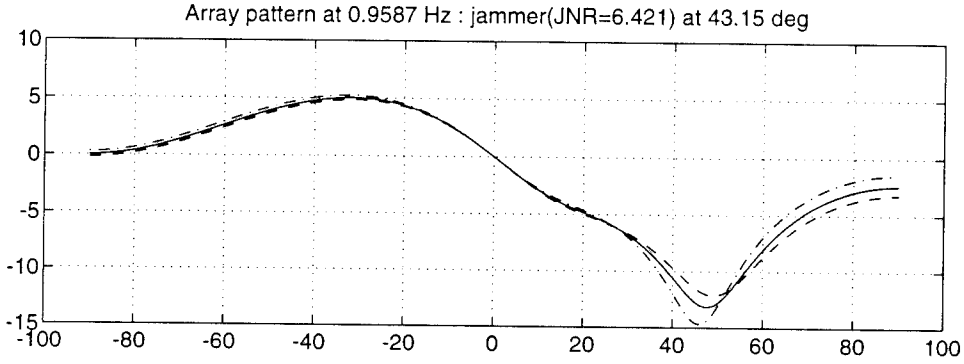


Figure 17: Pattern cuts for 3 sensor, 2 tap full rank optimal solution

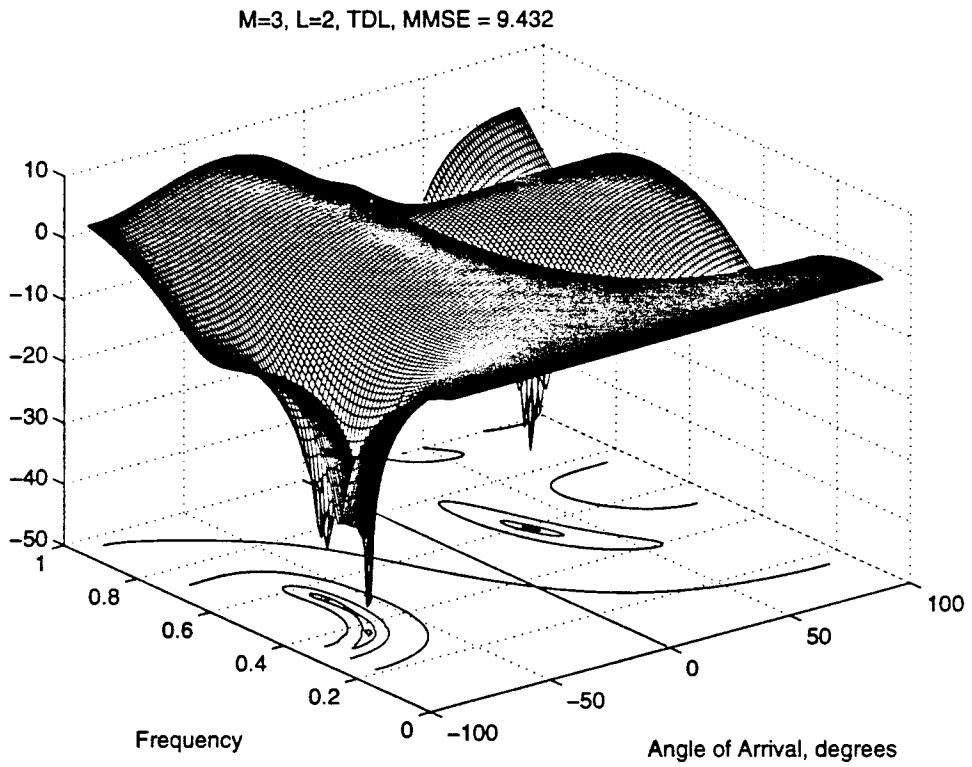


Figure 18: Complete pattern for 3 sensor, 2 tap full rank optimal solution

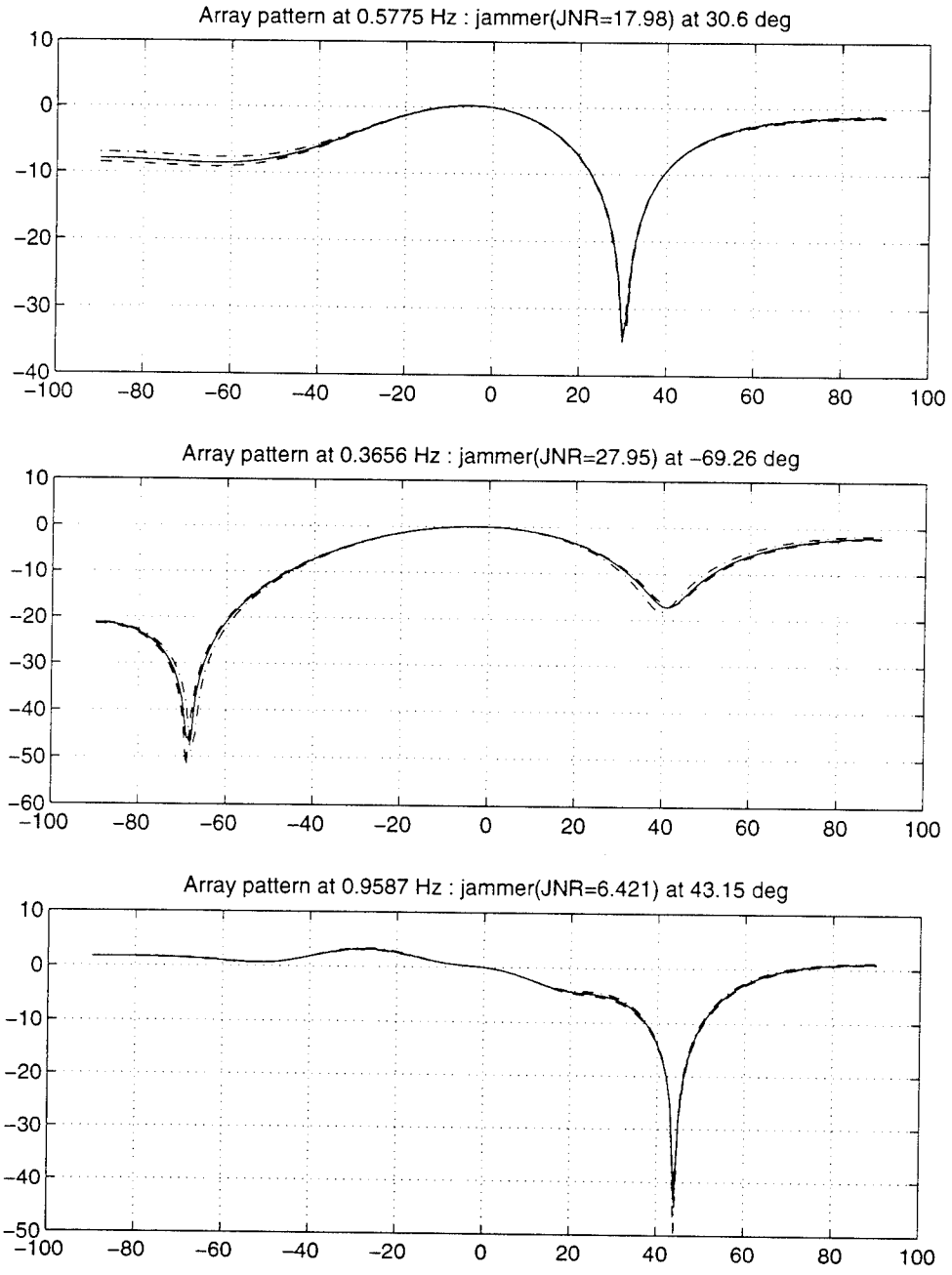


Figure 19: Pattern cuts for 5 sensor, 2 tap filterbank optimal solution

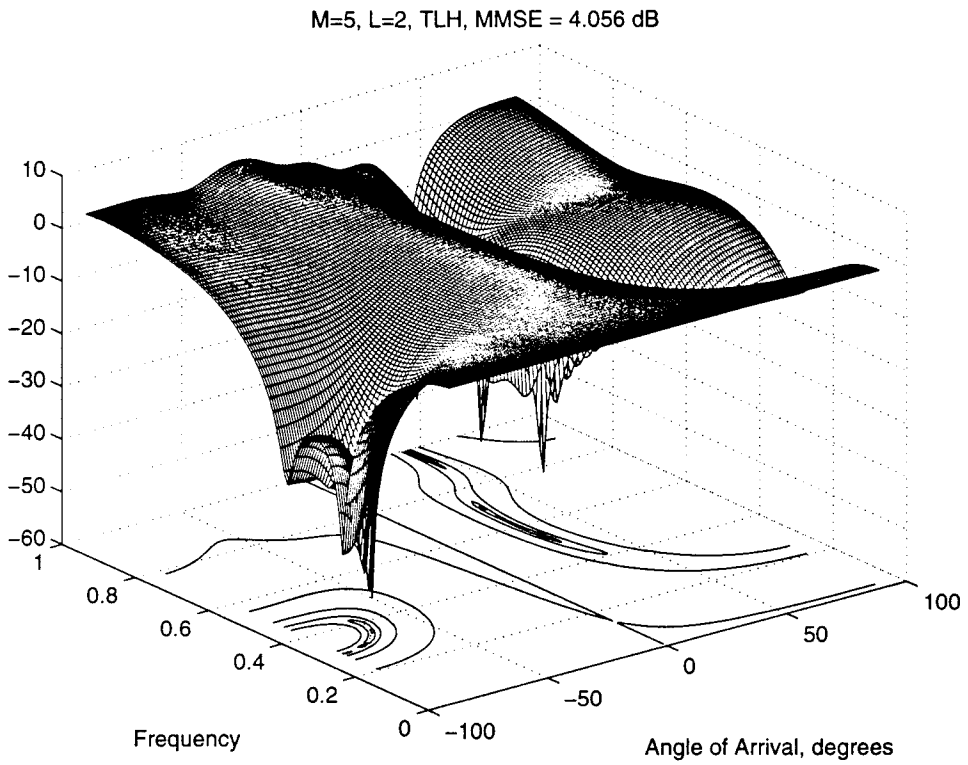


Figure 20: Complete pattern for 5 sensor, 2 tap filterbank optimal solution

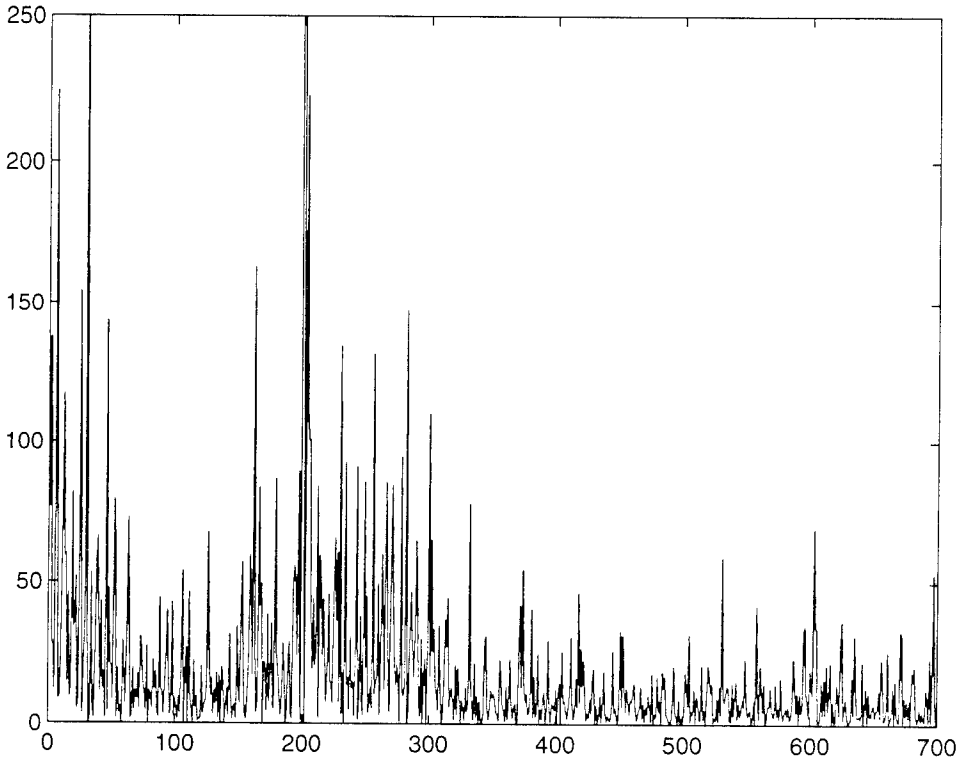


Figure 21: Learning curve for time-varying filterbank.

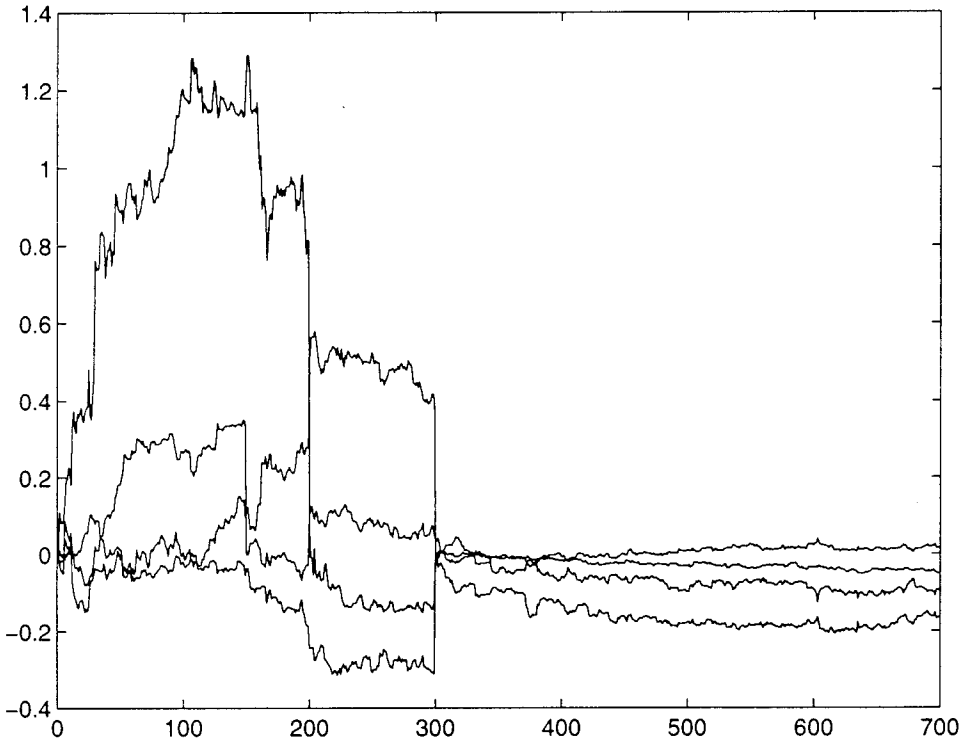


Figure 22: Trajectory for real parts of adaptive weights.

References

[1] D.J. Chapman, "Partial adaptivity for the large array," *IEEE Trans. AP*, Vol. 24, pp.685-696, Sept 1976.

[2] O.L. Frost, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, Vol. 60, No. 8, pp.926-935, August 1972.

[3] L.J. Griffiths and C.W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas and Propagation*, Vol. 30, No. 1,pp.27-34, January 1982.

[4] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, New Jersey, Prentice Hall, 1993.

[5] J.S. Goldstein, M.A. Ingram, E.J. Holder and R.N. Smith, "Adaptive subspace selection using subband decompositions for sensor array processing," *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Adelaide, South Australia, April 1994.

[6] J.S. Goldstein, M.A. Ingram, and E.J. Holder, "Adaptive space-time processing for radar receive arrays with two-dimensional subband decompositions," Proc. IEEE National Radar Conf., pp. 206-211, Atlanta, GA, March 1994.

[7] S.S. Narayan, A.M. Peterson and M.J. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. ASSP*, Vol. 31, No. 3, pp. 609-615, June 1983.

[8] T.T. Ma and L.J. Griffiths, "A solution space approach to achieving partially adaptive arrays," *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, New York, April 1988, pp. 2869-2872.

[9] B.D. Van Veen and R.A. Roberts, "Partially adaptive beamformer design via output power minimization," *IEEE Trans. ASSP* Vol. 35, pp. 1524-1532, Nov. 1987.

[10] B.D. Van Veen, "Improved power minimization based partially adaptive beamformer design," *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, New York, April 1988, pp. 2973-2976.

[11] B.D. Van Veen, "An analysis of several partially adaptive beamformer designs," *IEEE Trans. ASSP*, Vol. 37, pp. 192-203, Feb. 1989.

[12] B.D. Van Veen, "Optimization of quiescent response in partially adaptive beamformers," *IEEE Trans. ASSP* Vol. 38, No. 3., pp. 471-477, March 1990.

[13] S. Haykin and A. Steinhardt, eds., *Adaptive Radar Detection and Estimation*, Wiley, New York, 1992.

[14] S-J. Yu and J-H Lee, "Design of partially adaptive array beamformers based on information theoretic criteria," *IEEE Trans. Antennas and Propagation*, Vol. 42, No. 5, May 1994.

[15] G. H. Golub and C. F. Van Loan, *Matrix Calculations*, The Johns Hopkins University Press, Baltimore MD, 1983.

[16] B. Farhang-Boroujeny and S. Gazor, "Selection of orthonormal transforms for improving the performance of the transform domain normalised LMS algorithm," *IEE Proceedings-F*, Vol. 139, No.5, pp. 327-335, October 1992.

[17] J.C. Lee and C.K. Un, "Performance of transform-domain LMS adaptive algorithms," *IEEE Trans. ASSP*, Vol. 34, pp. 499-510, 1986.

[18] J.C. Lee and C.K. Un, "Performance of frequency-domain block LMS adaptive digital filters," *IEEE Trans. CS*, Vol. 36, pp. 173-189, 1989.

Rome Laboratory

Customer Satisfaction Survey

RL-TR-_____

Please complete this survey, and mail to RL/IMPS, 26 Electronic Pky, Griffiss AFB NY 13441-4514. Your assessment and feedback regarding this technical report will allow Rome Laboratory to have a vehicle to continuously improve our methods of research, publication, and customer satisfaction. Your assistance is greatly appreciated.
Thank You

Organization Name:_____(Optional)

Organization POC: _____(Optional)

Address:_____

1. On a scale of 1 to 5 how would you rate the technology developed under this research?

5-Extremely Useful 1-Not Useful/Wasteful

Rating_____

Please use the space below to comment on your rating. Please suggest improvements. Use the back of this sheet if necessary.

2. Do any specific areas of the report stand out as exceptional?

Yes___ No_____

If yes, please identify the area(s), and comment on what aspects make them "stand out."

3. Do any specific areas of the report stand out as inferior?

Yes___ No___

If yes, please identify the area(s), and comment on what aspects make them "stand out."

4. Please utilize the space below to comment on any other aspects of the report. Comments on both technical content and reporting format are desired.

☆U.S. GOVERNMENT PRINTING OFFICE: 1995-610-126-20036